



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/017,183	12/07/2001	Coach Wei	NW-102	5883
24227	7590	10/23/2006	EXAMINER	
EMC CORPORATION OFFICE OF THE GENERAL COUNSEL 176 SOUTH STREET HOPKINTON, MA 01748				ZHEN, LI B
			ART UNIT	PAPER NUMBER
			2194	

DATE MAILED: 10/23/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/017,183	WEI, COACH
	Examiner	Art Unit
	Li B. Zhen	2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 30 June 2006.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-18 and 22-25 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-18 and 22-25 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

*WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER*

Attachment(s)

- 1) Notice of References Cited (PTO-892)
- 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

- 4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) Notice of Informal Patent Application
- 6) Other: _____.

DETAILED ACTION

1. Claims 1 – 18 and 22 – 25 are pending in the current application.

Response to Amendment

2. Examiner acknowledges receipt of the supplemental Application Data Sheet [06/30/2006] providing the inventors' addresses.

3. The amendment to the abstract overcomes the objection to the abstract presented in the Non-Final Office Action dated 04/05/2006.

4. The amendment to the specification to delete line 1 page 2 through line 8 page 5 overcomes the objection under 35 U.S.C. 132(a).

5. The amendment to the specification to delete lines 4-6 on page 10 overcomes the objection to the disclosure for containing embedded hyperlinks.

6. The amendments to the claims overcome the 35 U.S.C. 101, 112 second paragraph rejections and claim objections presented in the Non-Final Office Action dated 04/05/2006.

Response to Arguments

7. Applicant's arguments filed 06/30/2006 have been fully considered but they are not persuasive. In response to the Non-Final Office Action dated 04/05/2006, applicant argues:

(1) Bahrs teaches the opposite of a central processing center, it can not teach or suggest "where the business logic of the application is running on a backend server." As a result, Bahrs can not suggest the business logic running on a backend server as the application logic in Bahrs is rendered on the client device. [p. 13];

(2) Applicants also respectfully disagree with Examiner's assertion that Bahrs discloses a "replacing the GUI API with a network aware GUI API running on a backend server which translates the application's presentation layer information into pre-determined format based messages which describe the Graphical User Interface." [p 14]

(3) Applicants also respectfully disagree with Examiner that Bahrs re-implements any API, or implements one that is network aware instead of being local machine centric as traditional GUI APIs. [p. 14]

(4) Since Bahrs uses the existing API, the Bahrs reference does not disclose re-implementing. Further, neither in this section or the entire patent does Bahrs teach or suggest implementing any sort of API nor even defining one to be network aware. [pp. 14 – 15]

(5) Applicants assert that Examiner misunderstands the definition of an application's presentation layer, which includes the structure of the GUI that is used to present information to the user. The presentation layer does not describe the data

information displayed to the user, but the conveying vehicle such as text boxes, check boxes, as well as, but not limited to, the general layout of the display. [p.15]

(6) The portion of the Bahrs reference Examiner asserts covers "sending such messages to the client via a network" (Column 41 line 66- Column 42 line 19) refers only to data to be displayed not data that describes the presentation layer of the application. As such, Bahrs does not teach or suggest transmitting data that describes the application's presentation layer. [p. 16]

(7) Applicants respectfully disagree that Bahrs, in general or at Examiner's quote, teaches or discloses a method in which the API "delivers the user experience for that device according to the capability of the specific client device." [p. 16]

(8) Applicants respectfully disagree that Bahrs discloses "processing the user input and client-side events on the backend server." [p. 16]

(9) The current invention however, uses a different approach by taking an application, developed for any client system, and providing an API capable of translating this graphic display to one for any client machine. Instead of Bahrs method of tailoring each application, the current invention simply creates an API that translates each application's GUI to the capabilities of each respective client. [p. 17]

(10) Bahrs does not suggest replacing the API that is the Java Foundation class rather only suggests augmenting it. [pp. 17 – 18]

(11) Bahrs does not say that the program is based on the operating system's API rather that it is based on the Java API. [p. 18]

(12) Bahrs only discloses using this method for a Java API. It mentions deploying a specially tailored application developed using the java design patterns on different platforms but does not mention the Palm API or the Windows CE API or using a non- Java API [p. 18]

(13) First, Bahrs teaches writing a client side program for each client using his invention as a toolbox to speed development. By definition, using a toolkit to augment development of a program is not the same as developing a program solely with an API. [p. 19]

(14) As Bahrs does not suggest replacing the API, it can not suggest that the API to be replaced is the AWT. [p. 19]

(15) Bahrs does not use HTTP as the predetermined protocol to send a description of the presentation layer as Bahrs never discloses communication of the presentation layer of a GUI [pp. 19 – 20]

(16) Bahrs does not use HTTPS as the predetermined protocol to send a description of the presentation layer of the GUI client. [p. 20]

(17) Applicants use of the network, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the network to send user data between server and client. [pp. 20 – 21]

(18) Applicants use of the XML, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses XML to send user data between server and client. [p. 21]

(19) Applicants use of the internet, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the internet to send user data between server and client. [p. 21]

(20) Applicants use of a local area network, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the local area network to send user data between server and client. [p. 22]

(21) Applicants use of a bandwidth-limited slow speed network, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the bandwidth-limited slow speed network to send user data between server and client. [p. 22]

(22) Examiner may have asserted the word mobile to be synonymous with wireless, but as stated in the preceding reference to the Bahrs specification, this is not the case. [pp. 22 – 23]

(23) Bahrs presenting different client types does not teach or disclose Claim 17 where the client device is selected from the group of workstations, desktops, laptops, PDAs, wireless devices, and other edge devices [p. 23],

(24) It does, at no point, clearly point out or infer that the client and server are one entity. [pp. 23 – 24]

(25) There is also no motivation to combine these two references. [pp. 25 – 26]

(26) Applicants use of the predetermined protocol, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the predetermined protocol to send user data between server and client. [p. 26]

(27) Applicants use of the predetermined messaging format which is proprietary, to send information to describe the application presentation layer, is wholly different than that of Bahrs which uses the predetermined messaging format which is proprietary to send user data between server and client. [p. 26]

As to argument (1), examiner respectfully disagrees and submits that the distributed processing system of Bahrs refers to a network of computers [Distributed data processing system 100 is a network of computers; col. 12, lines 3 – 16]. Bahrs does not disclose distribute processing across the devices in the system. For example, Bahrs discloses that distributed data processing systems involve data transfers between clients and servers. Finally, Bahrs clearly teaches business logic executing on a backend server [server side business logic, col. 31, lines 5 – 15 and col. 14, lines 23 – 36]. The business logic is executing of the server side and is not distributed across the devices in the system.

In response to argument (2), examiner respectfully disagrees and submits that Bahrs teaches replacing the GUI API with network aware GUI API [ViewController interface 3902 extends JTC interface 3904; col. 35, lines 45 – 54 and col. 44, line 13 – 50] running on a backend server [application containing the view controller may be located on the server; col. 36, line 65 – col. 37, line 15] which translates the application's presentation layer information [col. 47, line 63 – col. 48, line 15] into a predetermined format based messages [Object data may take various forms, such as Extensible Markup Language (XML), String, Hypertext Markup Language (HTML), key/value, Remote Method Invocation (RMI), J/XFS, RS232; col. 17, lines 25 – 39]

which describe a Graphical User Interface [col. 48, lines 40 – 60 and col. 53, lines 3 – 20]. Bahrs teaches the ViewController replaces the JTC interface by extending the JTC interface and the ViewController is running on a backend server. The object data contains data that describe a graphical user interface and Bahrs discloses that the object data may take various forms such as XML and HTML.

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., re-defining an API such that an application can be developed once in a standard manner and deployed multiple times through the use of the network-aware API, p. 14) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

In response to argument (3), examiner respectfully disagrees and submits that the claims do not require re-implementing any API. Bahrs teaches the ViewController interface extends the JTC interface and the ViewController is network aware because the ViewController is in communication with the JTC client [col. 47, line 63 – col. 48, line 15] and listens for input data from the JTC client [col. 16, lines 12 – 30].

In response to argument (4), examiner respectfully disagrees and submits that the claims do not require re-implementing any API. As to a network aware API, see the response to argument (3) above.

As to argument (5), examiner respectfully disagrees and submits that the claims do no require the definition of an application's presentation layer to include the structure

of the GUI that is used to present information to the user. In addition, Bahrs discloses using the object data to refresh the display user interface controlled by the ViewController; thus, the object data correspond to the application's presentation layer. Finally, Bahrs also discloses the ViewController can be used to generate alternate views [col. 36, line 65 – col. 37, line 16]. Therefore, Bahrs teaches application's presentation layer.

As to argument (6), examiner respectfully disagrees and submits that Bahrs discloses using the object data to refresh the display user interface controlled by the ViewController; thus, the object data correspond to the application's presentation layer. Finally, Bahrs also discloses the ViewController can be used to generate alternate views [col. 36, line 65 – col. 37, line 16]. Therefore, Bahrs teaches application's presentation layer.

As to argument (7), examiner respectfully disagrees and submits that Bahrs discloses "delivers the user experience for that device according to the capability of the specific client device" because Bahrs teaches that mechanism for creating the HTML view is application dependent/screen dependent [col. 37, line 50 – 67]. Therefore, the user experience for that device is based on the capability of the specific client device [screen dependent].

As to argument (8), examiner respectfully disagrees and submits that Bahrs discloses processing user input and client-side events on the backend server. Bahrs teaches processing user input [User inputs occur on the components, containers and beans and user input is in the form of an AWTEvent; col. 15, line 52 – col. 16, line

12] and processing client-side events on the backend server [deliver the information to the server's service for processing; col. 16, line 56 – col. 17, line 15]

In response to argument (9), examiner respectfully disagrees and submits that the claims do not specifically recite providing an API capable of translating this graphic display to one for any client machine. In addition, Bahrs teaches that mechanism for creating the HTML view is application dependent/screen dependent [col. 37, line 50 – 67 and col. 36, line 65 – col. 37, line 16].

As to argument (10), examiner respectfully disagrees and submits that Bahrs teaches the ViewController replaces the JTC interface by extending the JTC interface and the ViewController is running on a backend server [col. 35, lines 45 – 54 and col. 44, line 13 – 50]. The ViewController is a new class that is an extension of the JTC interface and the creation of the ViewController replaces the JTC interface. Therefore, Bahrs teaches replacing the API that is the Java Foundation class.

In response to argument (11), examiner disagrees and notes that Bahrs teaches Java run in conjunction with the operating system and provides calls to the operating system from Java programs [col. 13, lines 43 – 60]. The Java programs require function calls to the operating system; therefore the program is based on the operating system's API.

As to argument (12), examiner disagrees and notes that Bahrs teaches Java run in conjunction with the operating system and provides calls to the operating system from Java programs [col. 13, lines 43 – 60]. The Java programs require function calls to the operating system; therefore the program is based on the operating system's API. In

addition, Bahrs teaches the data processing system may be a PDA [col. 14, lines 1 – 17]; therefore the operating systems include Palm and Windows CE.

In response to argument (13), examiner respectfully disagrees and notes that Bahrs does not teach using a toolbox. In fact, examiner was unable to locate any reference to a toolbox in the Bahrs reference. Bahrs discloses using the JTC to provide for concurrent development of the client [col. 15, lines 25 – 52] and the JTC is a Java implementation for building client applications [col. 15, lines 25 – 52]; therefore, Bahrs disclose client program development using the Java API.

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., "a development once and deployment to multiple clients" and "creating a program using solely the Java API") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

As to argument (14), examiner respectfully disagrees and submits that Bahrs teaches the ViewController replaces the JTC interface by extending the JTC interface and the ViewController is running on a backend server [col. 35, lines 45 – 54 and col. 44, line 13 – 50]. The ViewController is a new class that is an extension of the JTC interface and the creation of the ViewController replaces the JTC interface. Therefore, Bahrs teaches replacing the API that AWT [col. 14, lines 36 – 65].

In response to argument (15), examiner respectfully disagrees and submits that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] and the HTTP protocol for communication [col. 15, lines 26 – 52].

As to argument (16), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] and the HTTPS protocol for communication [col. 15, lines 26 – 52].

As to argument (17), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] over a network [col. 12, lines 16 – 43].

As to argument (18), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] using XML messages [col. 37, line 50 – 67].

In response to argument (19), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] over the Internet [col. 12, lines 16 – 43].

As to argument (20), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] over a local area network [col. 12, lines 16 – 43].

As to argument (21), examiner disagrees and notes that Bahrs teaches the communication of presentation layer of a GUI [see the response to argument (5) above] over a bandwidth-limited slow speed network [col. 1, line 58 – col. 2, line 15].

In response to argument (22), examiner respectfully disagrees and submits that Bahrs teaches a wireless network [Mobile Channel support; col. 15, lines 25 – 54] and RS232 devices, which includes a wireless standard for communication between devices. Therefore, Bahrs teaches a wireless network.

In response to argument (23), examiner respectfully disagrees and submits that Bahrs disclose the client device is selected from the group of workstation, desktops [col. 16, lines 46 – 57], laptops, PDAs [col. 14, lines 1 – 23], wireless devices [see response to argument (22) above] and other edge devices. It is also noted that the claim requires the device to be one selected from the group of workstations, desktops, laptops, PDAs, wireless devices, and other edge devices. Therefore, Bahrs teaches applicant's invention as claimed.

As to argument (24), examiner respectfully disagrees and submits that Bahrs discloses the client and server are one entity [business logic and data can be physically located on any machine; col. 31, lines 5 – 15].

In response to argument (25), the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, the motivation to combine Bahrs and Rennard can be found in col. 7, line 63 – col. 8, line 23 and col. 8, lines 21 – 39 of Rennard.

As to argument (26), examiner disagrees and notes that Bahrs as modified by Rennard teaches the communication of presentation layer of a GUI [see the response to argument (5) above] using predetermined protocol [col. 6, line 57 – col. 7, line 2 of Rennard].

As to argument (27), examiner disagrees and notes that Bahrs as modified by Rennard teaches the communication of presentation layer of a GUI [see the response to argument (5) above] using predetermined proprietary messaging format [col. 6, line 57 – col. 7, line 2 of Rennard].

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –
(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

9. **Claims 1-8, 11, 13-18 and 22-25 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,901,554 to Bahrs et al. [hereinafter referred to as Bahrs, cited in the previous office action].**

10. As to claim 1, Bahrs teaches a method for delivering applications over a network in which the business logic of the application [business logic; col. 31, lines 5 – 15 and

col. 14, lines 23 – 36] is running on the backend server [a server 104; col. 12, lines 16 – 43; server side business logic, col. 31, lines 5 - 15], the method comprising the steps of:

having the application invoke a GUI API to present the application's user interface [a client browser invokes a URL submit, the web server obtains the request and passed control to a servlet. The servlet obtains a key/value pair list of values entered in the HTML client. This list is passed to the ViewController alternate view being displayed; col. 38, lines 1 – 19];

replacing the GUI API with network aware GUI API [ViewController interface 3902 extends JTC interface 3904; col. 35, lines 45 – 54 and col. 44, line 13 – 50] running on a backend server [application containing the view controller may be located on the server; col. 36, line 65 – col. 37, line 15] which translates the application's presentation layer information [col. 47, line 63 – col. 48, line 15 and col. 36, line 65 – col. 37, line 16] into a pre-determined format based messages [Object data may take various forms, such as Extensible Markup Language (XML), String, Hypertext Markup Language (HTML), key/value, Remote Method Invocation (RMI), J/XFS, RS232; col. 17, lines 25 – 39] which describe a Graphical User Interface [col. 48, lines 40 – 60 and col. 53, lines 3 – 20], event processing registries [data is passed via different events, such as ViewEvent 510, RequestEvent 522, and RequestEvent 526; col. 17, lines 25 – 39] and other related information [object handling placement of components will register as a listener for notifications to place objects on the screen; col. 24, lines 36 – 59], the presentation layer of the application in a high level, object level messages [col. 16, line 57 – col. 17, line 15];

sending such messages to the client device via a network [col. 41, line 66 – col. 42, line 19; col. 48, lines 40 – 60 and col. 53, lines 3 – 20];

processing the messages and rendering a user interface by a client-side program [If the major code for the TopEvent is message, then the message is displayed for the application (step 8418); col. 49, lines 25 – 33], which delivers a user experience for that device according to the capability of the specific client device [mechanism for creating the HTML view is application dependent/screen dependent; col. 37, line 50 – 67];

rendering the user interface on the client device [ViewController 502 basically provides a reusable GUI element; col. 15, line 52 – col. 16, line 13];

transmitting a plurality of user input and client-side events back to the server by the client-side program [col. 36, lines 17 – 28] via a predetermined protocol [col. 14, lines 36 – 65];

processing the user input and client-side events on the backend server [col. 26, lines 1 – 20 and col. 16, line 56 – col. 17, line 15], translating such events and inputs as if they were locally generated [ViewEvents generated in the ViewControllers 12302 being handled by the ApplicationMediator 12304 and translated into appropriate RequestEvents; col. 65, lines 23 – 41], and sending such translated events and inputs to the application for processing [RequestEvents are passed on to the destination 12308 via the transported 12306; col. 65, lines 23 – 41];

encoding and routing the output of the application to the client device using the predetermined messaging format [col. 16, line 57 – col. 17, line 15]; and

further processing the output by the client-side program to refresh the Graphical User Interface thereat [the return data may be sent to ViewController 502 to refresh the view displayed on the screen to the user; col. 16, line 57 – col. 17, line 15].

11. As to claim 2, Bahrs teaches the GUI API and the event processing API are Java Foundation Classes [col. 14, lines 36 – 65].

12. As to claim 3, Bahrs teaches the client-side program is a computer program based on Operating System's API, such as Windows API, or X Windows API [col. 34, lines 30 – 39 and col. 13, lines 43 – 60].

13. As to claim 4, Bahrs teaches the client-side program is a wireless device program written using the device's Operating System's API, such as Palm API and Windows CE API [col. 15, lines 26 – 52 and col. 14, lines 1 – 17].

14. As to claim 5, Bahrs teaches the client-side program is Java program written using Java API [col. 14, lines 36 – 65 and col. 15, lines 25 – 52].

15. As to claim 6, Bahrs teaches the JAVA API is AWT, Personal Java, Java 2 Micro Edition based GUI API or Java Swing [col. 14, lines 36 – 65 and col. 35, lines 45 – 54 and col. 44, line 13 – 50].

16. As to claim 7, Bahrs teaches the predetermined protocol is HTTP [JTC has natural support for multiple protocols, such as, for example IIOP, RMI, Sockets, HTTP, HTTPPs, and Files; col. 15, lines 26 – 52].

17. As to claim 8, Bahrs teaches the predetermined protocol is HTTPS [JTC has natural support for multiple protocols, such as, for example IIOP, RMI, Sockets, HTTP, HTTPPs, and Files; col. 15, lines 26 – 52].

18. As to claim 11, Bahrs teaches the predetermined messaging format is based on XML [col. 17, lines 25 – 38 and col. 37, line 50 – 67].

19. As to claim 13, Bahrs teaches the network is the Internet [col. 12, lines 16 – 43].

20. As to claim 14, Bahrs teaches the network is a local area network [col. 12, lines 16 – 43].

21. As to claim 15, Bahrs teaches the local area network is a bandwidth-limited slow speed network [col. 1, line 58 – col. 2, line 15].

22. As to claim 16, Bahrs teaches the network includes a wireless network [col. 15, lines 25 – 52].

23. As to claim 17, Bahrs teaches the client device is selected from the group consisting of workstations, desktops, laptops, PDAs, wireless devices and other edge devices [col. 15, lines 25 – 52].

24. As to claim 18, Bahrs teaches the server and the client device are combined into one entity [col. 17, lines 61 – 67 and col. 31, lines 5 – 15].

25. As to claim 22, Bahrs teaches a system for distributing an application [col. 14, lines 23 – 36] including at least a server [a server 104; col. 12, lines 15 – 45], at least a client device [clients 108, 110, and 112; col. 12, lines 16 – 43], and a communication means [network 102; col. 12, lines 16 – 45], the system comprising:

 a presentation layer of the application [ViewController; col. 15, line 52 – col. 16, line 12] written using a server-side API [col. 19, lines 12 – 30] based network programming model [col. 28, lines 42 – 67];

 a business logic layer of the application [business logic; col. 31, lines 5 – 15 and col. 14, lines 23 – 36] and a data layer of the application [data model; col. 35, line 57 – col. 36, line 6] both of which are written with the server-side API and running on the server [a server 104; col. 12, lines 16 – 43; server side business logic, col. 31, lines 5 – 15]; and where

 the server-side API having a supporting infrastructure that sends [Object data may take various forms, such as Extensible Markup Language (XML), String, Hypertext Markup Language (HTML), key/value, Remote Method Invocation (RMI), J/XFS, RS232;

col. 17, lines 25 – 39] the application's user interface information [col. 47, line 63 – col. 48, line 15] to a client device for presentation [col. 48, lines 40 – 60 and col. 53, lines 3 – 20], handles communications problems [col. 43, lines 15 – 36], renders the application's user interface [ViewController 502 basically provides a reusable GUI element; col. 15, line 52 – col. 16, line 13] and dispatches necessary user input events back to the server for processing [col. 18, line 63 – col. 19, line 13].

26. As to claim 23, an apparatus for distributing an application over a network [col. 14, lines 23 – 36] where the apparatus includes:

 a server [a server 104; col. 12, lines 15 – 45];
 a client device [clients 108, 110, and 112; col. 12, lines 16 – 43];
 a network communication means [network 102; col. 12, lines 16 – 45];
 a network based API module that is used to transparently replace the API on which the application was developed [ViewController interface 3902 extends JTC interface 3904; col. 35, lines 45 – 54 and col. 44, line 13 – 50];
 a first means for running an application of the plurality of applications where a business logic [business logic; col. 31, lines 5 – 15 and col. 14, lines 23 – 36] of the application runs on the server [a server 104; col. 12, lines 16 – 43; server side business logic, col. 31, lines 5 - 15];
 a second means for replacing the API of each of the plurality of applications with the network based API [Interfaces extending JTC are ViewController, ApplicationMediator, and Destination; col. 44, lines 13 – 51] so that each of the

applications' logic runs on the server [application containing the view controller may be located on the server; col. 36, line 65 – col. 37, line 15];

a third means for using the network based API to create a display for an application on the client device [ViewController 502 basically provides a reusable GUI element; col. 15, line 52 – col. 16, line 13];

a fourth means for transferring the user interactions on the client device to the server [col. 18, line 63 – col. 19, line 13], calculating the appropriate response to the input [deliver the information to the server's service for processing; col. 16, line 56 – col. 17, line 15], and transmitting the appropriate response to the client machine [response data will be returned to the Transporter 524 in a RequestEvent; col. 16, line 56 – col. 17, line 15];

a fifth means for updating the display of the application on the client device based on the responses from the server [return data may be sent to ViewController 502 to refresh the view displayed on the screen to the user; col. 16, line 56 – col. 17, line 15].

27. As to claim 24, Bahrs teaches the application code is not modified when distributing the application [col. 14, lines 23 – 36] and the application code is not distributed to the client device [business logic and central data management of an application should be separated out from the JTC application; col. 31, lines 5 – 15].

28. As to claim 25, Bahrs teaches distributing a plurality of pre-existing applications [col. 14, lines 23 – 36].

Claim Rejections - 35 USC § 103

29. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

30. **Claims 9, 10 and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bahrs in view of U.S. Patent No. 6,615,131 to Rennard et al. [hereinafter referred to as Rennard, cited in the previous office action].**

31. As to claim 9, Bahrs teaches wireless devices [col. 15, lines 26 – 52] and multiple protocols [col. 15, lines 26 – 52] but does not specifically disclose the WAP protocol.

However, Rennard teaches Java user interfaces [col. 8, line 64 - col. 9, line 37] and the WAP protocol [Wireless Application Protocol; col. 7, line 64 – col. 8, line 13].

It would have been obvious to a person of ordinary skill in the art at the time of the invention to apply the teaching of the WAP protocol to the invention of Bahrs because the Wireless Markup Language (WML) in the Wireless Application Protocol includes navigation and event-handling models that allow an author to specify the processing of user agent events [col. 7, line 63 – col. 8, line 23 of Rennard]. In addition, the WAP protocol allows the creation of a WML foundation class that reduces the

amount of code that must be written to create a WML deck [col. 8, lines 21 – 39 of Rennard].

32. As to claim 10, Bahrs as modified teaches the predetermined protocol is proprietary [col. 6, line 57 – col. 7, line 2 of Rennard].

33. As to claim 12, Bahrs as modified teaches the predetermined messaging format is proprietary [col. 6, line 57 – col. 7, line 2 of Rennard].

Conclusion

34. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

CONTACT INFORMATION

35. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on 571-272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2194

LBZ



WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER